静态路由转发实验实验报告

实验题目：

本实验要求学生在已有代码基础上，完善其中的TODO部分，实现路由器的IP查找转发、ARP请求和应答、ARP缓存管理、发送ICMP消息等功能。

实验说明：

arp.c # 发送ARP请求和应答

handle\_arp\_packet函数根据收到的arp请求种类进行相应的操作，arp\_send\_\*函数主要注意的就是根据要求填写条目。Iface\_send\_packet\_by\_arp会先去查找dst\_ip的mac地址，如果找不到则将其加入arpcache，如果找到了则填写ethernet头部并发包。

arpcache.c # ARP缓存相关操作

arpcache\_lookup就是去arpcache列表中查找对应ip的mac地址。

arpcache\_append\_packet先去查找是否有相同的ip和iface的条目，如果有那么就接在它后面，不用再额外发送arp request,只要等待之前发送的request回复就行。如果没有对应的条目，则创建新的条目并发送arp request。

arpcache\_insert区别于append，它是将Ip-mac对应插入arpcache中，每次插入都会去查看是否有等待着这条关系的packet，如果有则发送掉。

arpcache\_sweep是一个不断循环的函数，它会去检查arpcache，如果某个条目存在时间太长则删除，如果一个arp request很长时间也没收到回复，则重发，如果重发多次也没回应则发送icmp表示arp查询失败。

icmp.c # 发送ICMP数据包

icmp\_send\_packet对不同的type和code组装相应的icmp包并发送。

ip.c # IP前缀查找和发送IP数据包

longest\_prefix\_match以最长匹配原则查找对应ip的路由条目，ip\_send\_packet只负责发送icmp包，由icmp\_send\_packet调用，

ip\_forwarding.c # IP转发

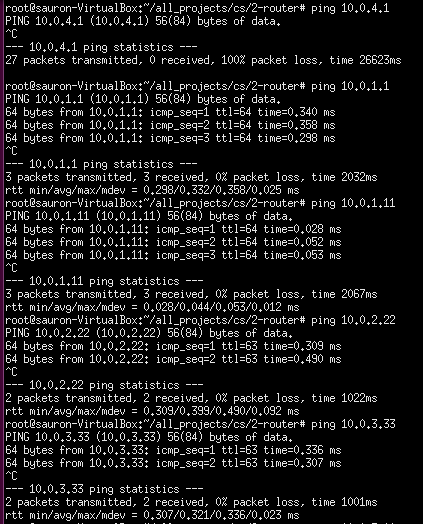
ip\_forward\_packet将收到的包进行继续转发，并且会去判断这个包的ttl以及checksum，如果有错则返回icmp信息。handle\_ip\_packet则会处理接受到的包，如果是发到本机的icmp echo request包则回复icmp echo reply，不然就ip\_forward这个包。

测试过程：

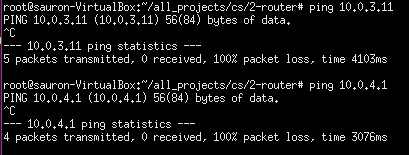
1. 运行sudo python router\_topo.py进入mininet界面，再运行xterm r1 h1打开两个终端，在r1中运行./router，在h1中ping各个不同ip。

结果如下：

对于在topo结构中的节点都能成功ping到。

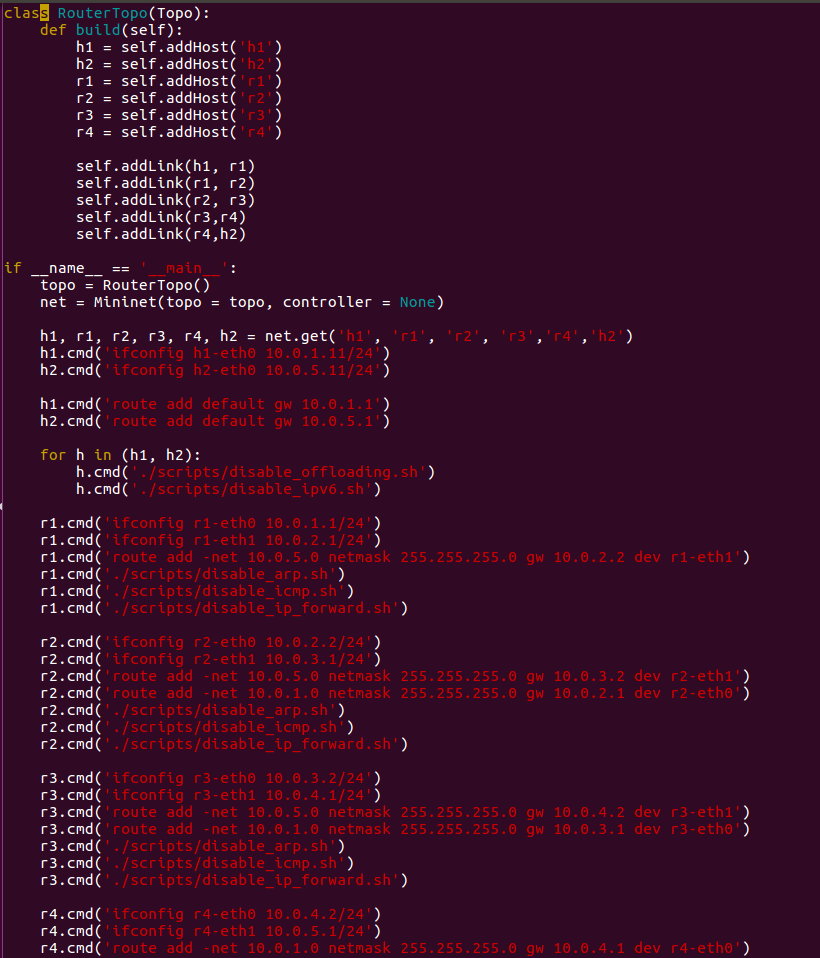


对于两个不能到达的节点，均ping失败。

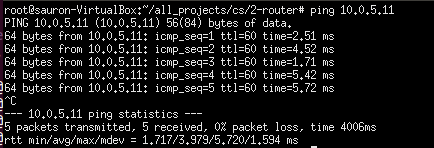


2.

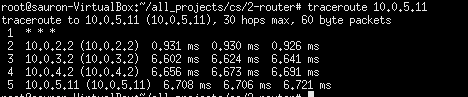
第二个实验我们自己构造了一个新topo，有两个host和4个router，具体结构如下：



进行ping操作：



进行traceroute操作



可见网络都顺利完成了任务。